1.数据分析与清理

数据的质量决定模型的上限。本部分通过分析数据发现不合理的数据并清除或修改。
sales_train.csv

```
print("物品最大价格",df_train['item_price'].max())
print("物品最低价格",df_train['item_price'].min())
```

```
物品最大价格 307980.0
物品最低价格 -1.0
```

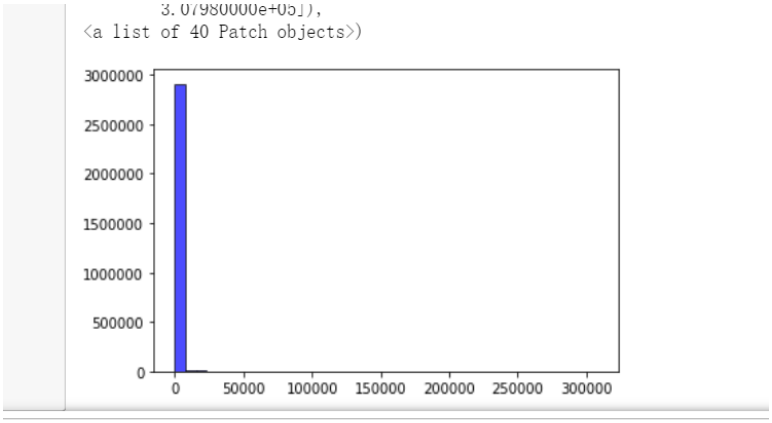
对于 item_price 字段，出现物品价格<0 的情况，显然是不合理的，因此将价格<1 的数据清除。

对于 item_cnt_day 字段，日商品售出<0 的情况，同样不合理，但是这类数据我们选择将<0 的归为 0，也就是这个商品日销售为 0.

```
print("最大购买数",df_train['item_cnt_day'].max())
print("最小购买数",df_train['item_cnt_day'].min())
```

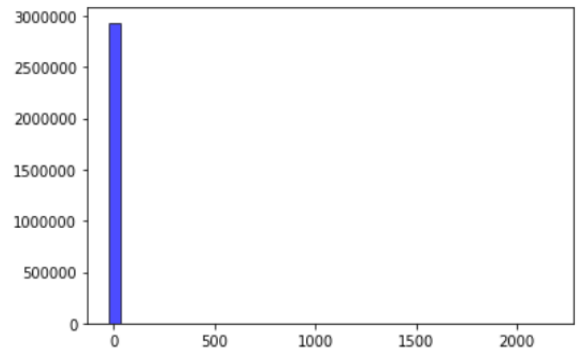
```
最大购买数 2169.0
最小购买数 -22.0
```

关于商品价格分布



大部分物品价格都不高，个别物品价格特别高

关于商品销量



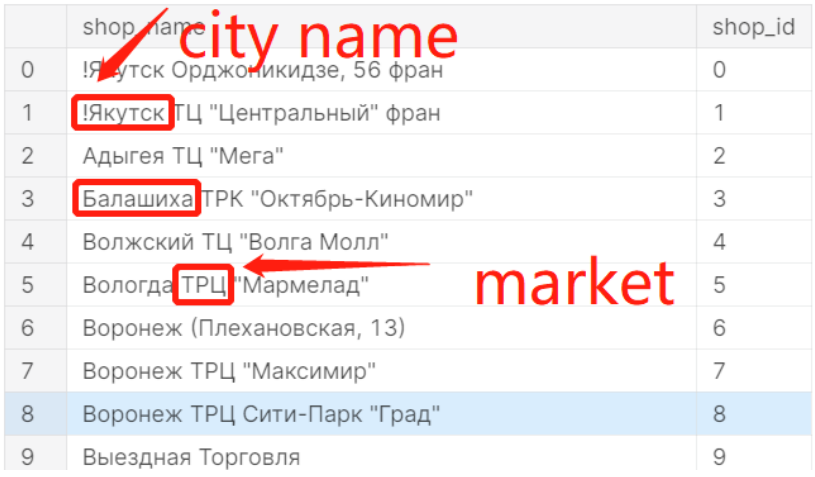
大部分商品的销售量都不高

(注：这两张图都可以在 EDA.ipynb 中复现)

2. 特征工程

2.1 shop

Shop 文件有两个字段，商品名和商品 id。商品名是俄文不太好懂，但通过谷歌翻译我们可以看到一下规律。



	shop_name	shop_id
0	!Якутск Орджоникидзе, 56 фран	0
1	!Якутск ТЦ "Центральный" фран	1
2	Адыгея ТЦ "Мега"	2
3	Балашиха ТРК "Октябрь-Киномир"	3
4	Волжский ТЦ "Волга Молл"	4
5	Вологда ТРЦ "Мармелад"	5
6	Воронеж (Плехановская, 13)	6
7	Воронеж ТРЦ "Максимир"	7
8	Воронеж ТРЦ Сити-Парк "Град"	8
9	Выездная Торговля	9

对于商品名，第一个单词为城市名，第二个字段为商店类型，隐含商店规模。因此可以将商品名称分离，构建两个特征，city，category。

2.2 cats

Cats 有两个字段，item_category_name，item_category_id。同样在 item_category_name 可以构造出两个特征。

	item_category_name	item_category_id	type_code
0	PC - Гарнитуры/Наушники	0	PC
1	Аксессуары - PS2	1	Игры
2	Аксессуары - PS3	2	Игры
3	Аксессуары - PS4	3	Игры
4	Аксессуары - PSP	4	Игры

从 1-4 行来看应该是游戏机，-前面就是游戏机的大类，后面就是小类别，将其分离出来作为特征。

2.3 items

t[45]:

	item_name	item_id	item_category_id
0	! ВО ВЛАСТИ НАВАЖДЕНИЯ (ПЛАСТ.) D	0	40
1	!ABBYY FineReader 12 Professional Edition Full...	1	76
2	***В ЛУЧАХ СЛАВЫ (UNV) D	2	40
3	***ГОЛУБАЯ ВОЛНА (Univ) D	3	40
4	***КОРОБКА (СТЕКЛО) D	4	40
...	...	...	...
22165	Ядерный титбит 2 [PC, Цифровая версия]	22165	31
22166	Язык запросов 1С:Предприятия [Цифровая версия]	22166	54
22167	Язык запросов 1С:Предприятия 8 (+CD). Хрустале...	22167	49
22168	Яйцо для Little Inu	22168	62
22169	Яйцо дракона (Игра престолов)	22169	69

22170 rows × 3 columns

对于 items，将（）内的内容作为一个特征，将[]内的内容作为一个特征。

2.4 时序特征的构建

对于月销量预测问题，过去的销量数据很重要，过去一个月的销量数据，过去两个月的销量数据，过去三个月销量数据，过去的平均月销量数据等。

类似的，此类特征有:

- 对于每个商店每个物品 过去一，二，三个月的月销量
- 每个月，所有商店所有商品平均销量
- 上个月，所有商店所有商品平均销量(可以反映销量趋势)
- 每个月 所有商店 某个商品平均销售数量，以及前一二三个月的销量
- 每个月 某商店的平均商品销售数量，以及前一二三个月的销量
- 每个月 某商店某商品的平均销售数量，以及前一二三个月的销量
- 每个月 每个商店 每个商品小类的平均销售数量，以及前一个月的销量
- 每个月 每个城市的平均商品销售数量，以及前一个月的销量
- 每个月 每个城市 每个商品的平均月销售量，以及前一个月的销量
- 每个月 每个商品的平均售价，以及前一二三个月的售价，以及差价(反映商品价格变化趋势)
- 每个月 每个商店的营业额，以及当月营业额与平均营业额变化比例

此外还有

- 月份特征，商品可能出现季节性销量变化
- 天数特征，商品可能出现月内销量变化波动
- 商品首卖时间，新商品可能在首卖暴涨，热度随着首发时间推移慢慢减弱。

3.模型构建

采用了 xgboost，lightgbm，catboost 三个树模型，树模型具有良好的可解释性和强大的拟合能力，在比赛中高频出现。

训练集采用前 33 个月的数据，验证集采用第 33 个月的数据，测试集为竞赛测试集。

单模效果:

Xgboost	lightgbm	Catboost
0.89273	0.90692	1.25006

融合效果:

融合的方式有很多，stackingh,blending 等。我采用的是最简单的结果取平均。（Xgboost+lightgbm+catboost）/3。

融合
0.8990

这个效果比单模 xgboost 效果稍差，其实是 catboost 效果太差拖了后腿。

4.代码使用说明

1. 请将数据集放在与代码同目录 data 文件夹下
2. 请确保环境配置以下依赖库:

- xgboost
- lightgbm
- catboost==0.5